



CUB USER GUIDE

CUSTOM UNITY BACK-END VERSION 1.2D

Copyright Notice

Copyright © 1998 Dataphone (UK) Limited. All rights reserved. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission from Dataphone (UK) Limited.

The information furnished herein is believed to be accurate and reliable. However, no responsibility is assumed by Dataphone (UK) Limited for its use, nor for any infringements of patents or other rights of third parties resulting from its use.

Trademarks Notice

Dataphone™ is a trademark of Dataphone (UK) Ltd.

All third party trademarks are the property of their respective holders.

Millenium Compliancy Notice:

Unity has been examined for year 2000 problems, and may be used on and beyond that date.

Dataphone (UK) Limited Offices

South East Regional Office (SERO)

Dataphone (UK) Limited
Unit 11
Chiltern Court
Asheridge Road
Chesham
Bucks HP5 2PX
United Kingdom

Tel:	+44 (0)1494 794800
Fax:	+44 (0)1494 794888
e-mail:	sales@dataphone.co.uk
Web:	www.dataphone.co.uk

East Midlands Regional Office (EMRO)

Dataphone (UK) Limited
Unit 11 The Courtyard
Whitwick Business Park
Stenson road
Coalville
Leicestershire LE67 4JP

Tel:	+44 (0)1530 811110
Fax:	+44 (0)1530 811112
e-mail:	sales@dataphone.co.uk
Web:	www.dataphone.co.uk

Table of Contents

Preface	1
About this Manual	1
Cross References	1
Typing Conventions.....	1
1 Introduction	3
1.1 What is Unity ?	3
1.2 Some Terminology	3
2 Diamond Infrastructure	5
2.1 Overview of Infrastructure	5
2.2 Double Server Fail-safe Operation (DSFO)	6
3 Unity Data-flow and Layers	7
3.1 General Description of the Data-flow	7
3.2 Data at the Switch	8
3.3 The Transport Layer	9
3.4 The Translation Layer	9
3.5 The Unity Database.....	9
3.6 The User Interfaces (Front-ends)	9
4 Transport Layer	11
4.1 CDR Transport	11
4.2 Events Data Transport.....	12
4.3 Statistical Data Transport	12
5 Translation Layer	13
5.1 CDR Data Translation	13
5.2 Events Data Translation	13
5.3 Statistical Data Translation.....	14
6 Data Storage and Access	15
6.1 How the Data are Stored	15
6.2 Brief Description of Unity Tables	15
7 Alarm Management	17
7.1 Alarm Sources	17
7.2 Dealing with Alarms.....	18
8 CDR Data File Processing	19
8.1 General Description of CDR Data Transfer	19
8.2 Directory Structures.....	21
8.2.1 The Directory: /products/unity	21
8.2.2 The Directory: /sites	23
8.2.3 The Directory: /export/home/ama.....	23

8.2.4 The Directory: /export/home/ama/fms/<site>	24
8.3 CDR File Naming Convention	24
8.4 The Processes.....	25
9 Maintenance	29
9.1 Manual File Management.....	29
9.2 Automatic Backup Procedure	29
9.3 Hard Disk Capacity.....	30
9.4 Manual Cleanup Tasks	30
10 Disaster Recovery	33
10.1 Mediation System Operation	33
10.2 Main Elements of the Disaster Recovery System	34
10.3 Outline of procedures	35
Glossary	37
Index	39

Table of Figures

Figure 1: Overview of Diamond Mediation Infrastructure	5
Figure 2: Unity Data-flow.....	8
Figure 3: Unity Data-flow with Existing Billing System for one of the Switches.....	20
Figure 4: CDR Data-flow for different Switch types.	21
Figure 5: Sequential Data Retrieval and Processing	26
Figure 6: Concurrent Data Retrieval and Processing	27

Table of Tables

Table 1: Site Names.....	23
Table 2: Glossary of Abbreviations	37
Table 3: Special Terms	37

Preface

About this Manual

This manual describes the operation of the Custom Unity Back-end for the company Diamond. The software has been specifically customised for Diamond and the manual reflects this customisation. This manual is not suitable for use with other versions of Unity. This manual is the sister manual of the UIMS User guide version 3.2G, which describes the Unity Front-ends, and can be used in conjunction with this manual.

Cross References

Cross-references are in the form "section number, section title". For example: Please see "1 Introduction"

Typing Conventions

The following typographical conventions are used throughout this guide:

- Dedicated file names, paths and commands etc. are formatted with the courier font.
For example: `directory/filename`
- Placeholders for mandatory information are enclosed in angular brackets. Optional parameters are enclosed in square brackets. Multi-valued parameters are separated by a `|` character.
For example: `ls <filename> [day|year|month]`.
- References to keyboard operations are in capital letters.
For example: CTRL, ALT, SHIFT, ENTER.
- Names of menu options and buttons as they appear on the screen are written in bold type.
For example: **Cancel**
- Some special terminology specific to the software is written in bold type, where it needs to be emphasised.
For example: **Alarm Details**

1 Introduction

This chapter provides you with a general introduction to Unity. It consists of the following sections:

- 1.1 What is Unity ?
- 1.2 Some Terminology

1.1 What is Unity ?

Unity is a complete solution for the mediation of CDR data (Call Detail Records). The Unity system handles the transfer of CDR data from every envisaged type of Switch to the central server where the data are stored, processed and sorted. The central server interfaces with one or more PC-based workstations running a variety of User Interfaces, either locally or remotely, where the overseeing maintenance personnel can easily view and interact with the various data as required. Unity is flexibly designed so that it can be tied into a company's existing mediation infrastructure. Every customer has different requirements, therefore the Custom Unity Back-end needs to be specifically tailored to suit individual customer requirements.

1.2 Some Terminology

To save confusion with terminology throughout this manual, some terms are now clearly defined:

DTX is a generic term for Digital Telephone Exchange. DTXs are more often referred to as *Switches*.

CDR data is a generic term for the operational data sent from the Switch that is ultimately used for billing purposes. CDR stands for Call Detail Record, and may be of many different formats.

Billing data is another term for CDR data.

An **Event** is an occurrence of something for which the Switch sends a warning message. The event may or may not be a problem. In many cases it is just informative. Events data are streamed to a specific port on the Switch.

An **Alarm** is a record of a *problematic* Event that has a non-zero priority level.

Statistical data is a generic term, and consists of operational measurements carried out at the Switch. Statistical data are streamed to a specific port on the Switch.

An **Operational Measurement** or **OM** is a particular record of Statistical Data.

A **Log** is a record of an event. The term 'Log' and 'Event' may often be interchanged.

A **Switch** is the common term for a Digital Telephone Exchange, or DTX.

2 Diamond Infrastructure

This chapter describes the Diamond mediation infrastructure and how Unity fits in with it. It consists of the following sections:

- 2.1 Overview
- 2.2 Double Server Fail-safe Operation (DSFO)

2.1 Overview of Infrastructure

Figure 1 provides an overview of the Diamond mediation infrastructure.

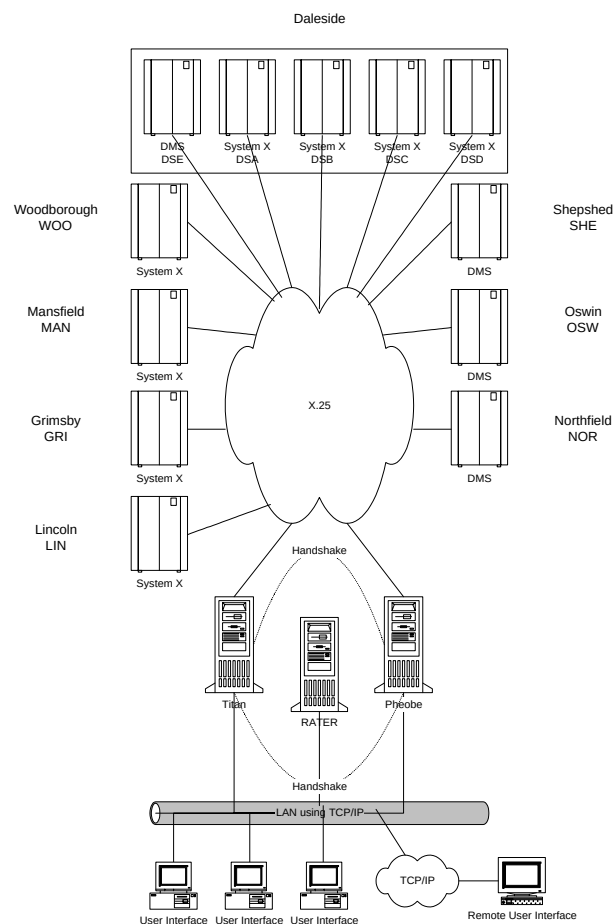


Figure 1: Overview of Diamond Mediation Infrastructure

The Diamond Mediation Infrastructure currently consists of eight sites with a total of twelve Switches, four of which are DMS Switches, the other eight being System-X Switches. The Unity Servers (Titan and Pheobe) are physically located at Daleside and Woodborough respectively. The billing system is an AS/400 based Subscriber Management System running CableMaster, and is based at Daleside, as is the Fraud Management System. The Woodborough and Daleside sites are connected to each other not only via the X.25 WAN, but also via the company LAN by means of NTL's own gateway. One or more PCs can be connected to the LAN to manage

Unity. This end-user software is called UIMS (Unity Interface System Manager). The Switches communicate with the Unity servers via the X.25 network.

The core of Unity is a Unix based server called Titan, the General Unity Server. Titan consists of the following components:

- A Unix server. This server is a powerful SPARC-based Sun machine, hardware version 5.2, running Solaris version 2.6, fitted with an X.25 Interface.
- An X.25 switch with integrated PAD (Packet Assembly/Disassembly), with associated cabling and line drivers.
- Unity back-end software, pre-installed and configured.
- Oracle database, version 8.0, pre-installed and configured.

Brief Description

The Switches are polled on a regular basis in order to transfer the CDR Data to the Titan, where they are checked, stored and processed. The Data is formatted for various requirements like the existing billing system (CableMaster), the Fraud Management System (FMS) and the Oracle Database. Events data originating from the DMS Switches, are also collected and processed. This version of Unity supports the transfer and processing of CDR data for both Switch types, and Events data for the DMS Switch type only. The transfer and processing of Statistical Data is not currently supported for this version of Unity.

To maintain compatibility with the existing billing system, copies of the CDR data files are kept on the Unity Server and are reformatted for the existing CableMaster billing system. Raw files are collected via an NFS mount for RATER.

The Unity system also provides all other necessary operations like file-handling processes, such as backup and cleanup, as well as the required interfacing with other peripheral devices such as printers. This is explained in further detail in "9 Maintenance".

2.2 Double Server Fail-safe Operation (DSFO)

To ensure that the billing data retrieval operation is not affected by hardware failure, the retrieval part of the Unity system is designed to run on two servers, such that when one fails, the other can be invoked. The primary server (Titan) has a higher spec in terms of storage space and has Oracle-based Unity database installed. The secondary server (Pheobe) has the same retrieval architecture but no Unity database. Pheobe is also known as the Disaster Recovery server (DR). The two servers are connected over both the LAN (Ethernet LAN running NFS), and the X.25 network. A full description of the Disaster Recovery Procedure is given in chapter 10 Disaster Recovery.

3 Unity Data-flow and Layers

This section describes the flow of data from the Switch to Titan. It consists of the following sub-sections:

- 3.1 General Description of the Data-flow
- 3.2 Data at the Switch
- 3.3 The Transport Layer
- 3.4 The Translation Layer
- 3.5 The Unity Database
- 3.6 The User Interfaces (Front-ends)

3.1 General Description of the Data-flow

Figure 2 broadly depicts the flow of data from the Switch through Titan, to the User Interfaces. This data-flow diagram is instructive in the understanding of how Unity works, and also the model that is used for the mediation monitoring process as used by the Unity Interface Manager. The description in this chapter applies to the pure “Unity concept” , and therefore does not cover the CDR data management aspects for the SMS and FMS systems, which are described later in the manual. This chapter describes the ‘pure’ Unity concept, where all the data are stored in an Oracle database. It does not describe the existing CDR data transfer and processing system, where an AS400 billing system is used.

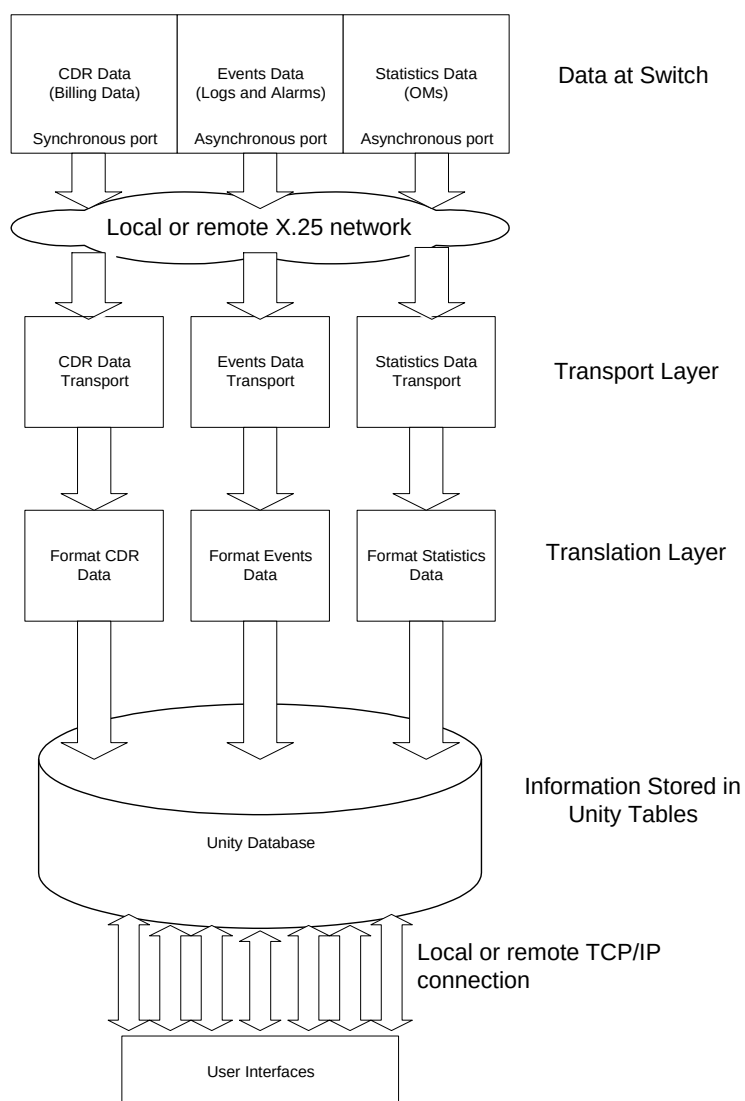


Figure 2: Unity Data-flow

3.2 Data at the Switch

The raw data originating at the Switch consist of three data components, namely CDR data (billing data), Events data (Logs and Alarms) and Statistical data (Operational Measurements, or OMs).

CDR Data

The CDR data exist in the form of one or more data files each comprising many call records. When the host polls the switch, new data files are downloaded from the Switch to Titan for processing.

Events Data

The Events data consist of logs and alarms that the switch generates. These data are not generally stored on the Switch, they are simply streamed via an asynchronous port as and when they are generated. It is up to the host to ensure that it retrieves the messages as they are sent. Events data are not currently processed for System-X Switches.

Statistical Data

Like the Events data, the Statistical data consists of a stream of data available at one of the Switch's asynchronous ports. This data stream consists of a list of operational measurements, and is accessed upon request by the host. Statistical data are not currently processed for System-X Switches.

3.3 The Transport Layer

The transport layer is responsible for the successful transfer of data from the Switch to the Unity Server, Titan. The Switches are connected to Titan via an X.25 network. The transport layer consists of the CDR transport module for copying the CDR data from the all the Switches to Titan, a PAD emulator for piping the Events data from the DMS Switches to Unity's Events Translation module. Currently, only the DMS Switches are supported for Events data. Statistical data are not currently supported.

Titan polls all the Switches for CDR data files, part simultaneously on a regular basis. The files are stored on Titan for further processing. This is explained in more detail later.

3.4 The Translation Layer

The translation layer is responsible for interpreting, organising and formatting the raw Switch data arriving at Titan. Again there is a module for each of the data components CDR data, Events data and Statistical data. The raw data are formatted, collated and organised into formats suitable for storage in the Unity Database, the SMS computer, and the FMS.

3.5 The Unity Database

The Unity Database stores all the data in meaningful tables in the Oracle database residing on Titan. The tables are then easily accessed as and when required by any sub-system or user interface.

3.6 The User Interfaces (Front-ends)

The Unity User Interfaces are user-friendly PC-based front ends that interface with the Unity Database. These front-ends present the data to the user in easy-to-understand manner. The main interface is called the Unity Interface Manager, or UIM for short. Through this interface you can access all other Unity front-ends.

The Unity User Interfaces are equipped with a comprehensive help system consisting of both the context sensitive **WhatsThis** type help as well as the usual main help system. Just about every label, field and button on the interfaces have context-sensitive descriptions associated with them. Please refer to the sister UIMS manual for a complete description.

4 Transport Layer

The Transport layer is responsible for the successful transfer of data from the Switches to the customer's computer system. It consists of three separate sub-systems, one for the CDR data, one for the Events data and one for the Statistical data. Each of these sub-systems is described in the following sections.

- 4.1 CDR Transport
- 4.2 Events Data Transport
- 4.3 Statistical Data Transport

4.1 CDR Transport

The CDR transport manages the electronic transfer of billing data between the Switch and Titan. Of the three data components, the CDR data transport is by far the most complex, as it must ensure absolute integrity and security of the billing data. If the connection is lost during a CDR transfer, for example, the transport mechanism must ensure that another attempt is made to retrieve the same data without loss or corruption of the data.

The Unity CDR Transport sub-system is a highly secure system and operates discretely in the background with a high degree of reliability. Minimal user intervention is required, thus technical staff are spared the burden of the system administration usually involved with this kind of *mediation*. Being a UNIX based system, Unity takes full advantage of the multi-tasking functionality offered by this operating system.

The CDR Transfer operation is invoked automatically at regular intervals. Unity polls the Switches and receives the raw CDR data in the form of data files containing the call records. This polling process is in synchronisation with the respective Switch. The exact polling sequence and intervals are functions of the Switch configurations. The data files are transferred from the Switches to the Unity server, over the X.25 network. These data files are stored in especially assigned directories. Copies of the files are made for backup purposes and reformatting for the SMS system and the FMS (Fraud Management System).

This process also generates its own detailed transfer log files, which are stored in a directory accessible to the user. This log file can be viewed directly with a standard ASCII editor if required. The most convenient way of viewing this log file is to copy it to a PC and view it using Notepad. Some of the essential information contained in the log file is also stored in the Unity database where it is subsequently used by the Unity User Interfaces to display the information (see the accompanying UIMS manual)". The detailed log file is dynamic and is being appended continually as the retrieval process continues.

The exact length of each transfer process depends on several factors such as the quantity of raw data to be retrieved, the data transfer rate, and the Switch response time, but is typically of the order of tens of minutes.

NOTE:

The maximum transfer rate limited to 64Kbps when using X.25 transfer. This limit is set by the X.25 specification, and is the absolute maximum data rate of an X.25 compatible synchronous port.

The CDR data transfer, is carried out automatically. There is normally no need for operator intervention. There are, however, a few basic administrative tasks that must be carried out from time to time, by a qualified system administrator. For further details of administrative tasks, see "9 Maintenance".

4.2 Events Data Transport

Currently, only DMS Switches are supported for Events data. The following description does not apply to System-X Switches.

Titan is fitted with an X.25 card for connection to the Switch via the X.25 network. Unlike for the CDR data, no special synchronisation is required between the Switch and Titan for Events data. The connection is relatively simple in that the log information is streamed out of the Switch in the form of data packets and sent directly to an assigned synchronous port on Titan, bypassing the external PAD. The internal PAD emulator simply disassembles the data packets (strips the unwanted addressing envelope) and pipes the useful ASCII data through to the translation layer.

The real-time log information streaming out of the Switch is continuous and is not stored. If the connection is broken, the log information is lost for the period of time that the connection is broken.

4.3 Statistical Data Transport

This is not currently implemented for NTL Midlands.

5 Translation Layer

The Translation Layer is responsible for translating the data arriving from the Switch into an intelligible format suitable for entering into the Unity database. There is a translation sub-system for each of the three data components, CDR data, Events data and Statistical data. Each of these sub-systems is described in the following sections.

- 5.1 CDR Data Translation
- 5.2 Events Data Translation
- 5.3 Statistical Data Translation

5.1 CDR Data Translation

When the CDR data transfer process for a Switch is finished, a copy of the raw CDR data files is made and placed in one of Unity's allocated directories. CDR data files are low level binary files, and can have various formats, depending on the Switch and its chosen format. Unity automatically recognises the data type and subsequently translates it into an intelligible format for further processing. After the CDR data transfer process has successfully copied these files to disk, it automatically initiates the CDR Translation processes, which are responsible for:

- a) Extracting the call data, collating them and placing them into the Unity tables.
- b) Formatting a copy of the CDR data file into a format suitable for the CableMaster billing system.
- c) Formatting the CDR data file into a format suitable for the FMS.

5.2 Events Data Translation

Currently, only DMS Switches are supported for Events data. The following description does not apply to System-X Switches.

The Events data transport layer pipes the Events data to the Events Data Translation Module where they are collated, sorted and placed in the Unity table ULOGS. The translation process for the DMS is now described in detail.

Event records arriving from the Switch are already assigned one of four priority levels, depicted by a prefix to the identifier, consisting of zero, one, two or three stars. If there are no preceding stars, the event is not interpreted as an alarm, but if there are any stars, the event is interpreted as an alarm, and allocated a default priority level.

For example:

TRK 138 is an informative event and is not registered as an alarm.

*DPNS 403 is a minor problem and is registered as an alarm with default priority level 7.

**TRK 102 is a major problem and is registered as an alarm with default priority level 4.

***TRK 103 is a critical problem and is registered as an alarm with default priority level 1.

These default priority levels can be altered by an administrative user if required. You do this by means of the Unity Events Interface, described in the accompanying UIMS manual.

When Unity is first installed, a lot of the Event records sent from the Switch will be new. Unity learns about new events, and every time a new event record is sent, Unity remembers it, along with its new priority level allocated by the administrative user. An unknown event becomes known when it is registered by the user via the Unity Events Interface.

When you view this log, using the Unity Events Interface, you can 'register' it by reassigning its priority level, whereby the log is thereafter 'known' to Unity.

Over the course of time, the amount of unknown logs will become scarcer, as they are registered, whilst the library of known logs will increase.

5.3 Statistical Data Translation

This is not currently implemented for NTL Midlands.

6 Data Storage and Access

This chapter describes the storage platform for the Unity data. It consists of the following sections:

- 6.1 How the Data are Stored
- 6.2 Brief Description of Unity Tables

6.1 How the Data are Stored

The raw CDR data files are stored in specially assigned directories on the Unix server, as described later in this manual. Once processed, they are backed up and marked for deletion. These data files are translated into formats suitable for the Unity database, the CableMaster billing system and the FMS.

Once the data have been translated for the Unity database, they are stored in the Unity Database in the form of tables. The platform chosen for the Unity database is a Unix based Oracle system. This platform offers a robust, efficient solution that is essential for handling the large amount of data storage and processing involved with telecommunications data mediation. These Unity tables contain all the information that any display, monitoring or billing process would ever need. Certain data, such as the raw CDR data files and ASCII log files are available in their raw forms on the hard disk. These may be used for other processes, as described later.

Interfacing with the Unity Database

The Unity Database concept offers a superior means of storing and accessing mediation data. All relevant data are conveniently stored in Unity tables in the Oracle database. This concept offers the following advantages:

1. The stored data format is completely independent from the Switch type and its raw data formats. In other words it offers complete isolation from back-end processes.
2. It offers complete design flexibility for display, monitoring and billing systems, which may now be developed independently of the Switch type and raw data formats.
3. The Unity database may be accessed across different platforms in real time, enabling any display, monitoring or billing process to be designed on and reside on any required platform (e.g. PC-based).
4. Oracle is a renowned powerful and professional solution, and can handle large quantities of data.

6.2 Brief Description of Unity Tables

This section contains a brief description of the main Unity tables, with which the Unity User Interfaces interact. Although not essential reading, this section may help to enhance your understanding the Unity system.

UALARMS (Alarms table)

UALARMS contains details of the alarm records. This table is the main source of information for the Unity Alarms Interface.

UBILLINGSTATUS (CDR status information)

UBILLINGSTATUS contains information about the CDR data files. This table is the source of information for the transfer view of the Unity CDR Interface.

UCDRDATA (Table containing CDR data)

UCDRDATA contains details of the actual CDR records. This data are used for the Unity CDR interface.

UNKNOWNLOGS (Table of Known Events)

This is a table of known events. New events can be memorized by Unity and stored in this table upon requirement. This table is accessed via the Unity Events Interface.

ULOGS (Table of Current Events)

This table contains details of the event records, sometimes known as logs.

URETENTION (Table containing record lifetimes)

This table is used by Unity's cleanup operation. It specifies the default number of days that a record should be retained before deletion, for each table.

USOURCE_ID (Table of alarm source IDs)

This table contains a list of the alarm source IDs with their descriptions.

USTATEVENT (Table of Statistical Events)

This table is used by the Unity Statistical Interface.

USWITCHLIST (List of Switches)

This table contains a list of the Switch ID numbers.

UTIMECHANGE (Clock change information)

This table contains details of time change information for internal use.

UTTICKET (Trouble-tickets definition)

This table contains details of the Trouble-ticket records. It is written to by the Unity Trouble-tickets interface.

UUNKNOWNLOGS (Table of Unknown Events)

This is a table of unknown events. It stores a list of Event records that have not yet been memorized by Unity. This table is accessed via the Unity Events Interface

UUSERS (List of Users)

This table contains a list of users and their security levels. It is accessed by the User Manager.

7 Alarm Management

A very important part of Unity is the Alarm Management System. It is conceivable that anything in any part of the system could have a problem at some stage, whether it be a problem at the Switch, the X.25 network, a module on Titan, or a user interface. This chapter describes the way Unity receives and processes alarms. It contains the following sections:

- 7.1 Alarm Sources
- 7.2 Dealing with Alarms

7.1 Alarm Sources

The alarm sources can be broadly split into three main categories:

1. Alarms generated from event records sent from the Switch (external alarms)
2. Alarms arising from the data transmission (transport alarms)
3. Alarms arising from Unity's modules, sub-systems or UIs. (internal alarms)

Every potential alarm source has been assigned an Alarm Source ID. This enables Unity to exactly identify the source and nature of the alarm, record the event in the Alarms table in the Unity database and subsequently display details about the alarm, including a full description and its priority. The Alarm Source ID is also used in the Unity Alarms Interface and the Unity Paging Interface. This is mostly used for technical support services, and will help Dataphone to quickly identify and rectify any problem.

Alarm types can be broadly categorised into three main groups:

External Alarms

A significant proportion of the alarms are produced by Unity's logging sub-system, where the Events originating from the Switch are processed. The logging sub-system translates these Events and stores them in tables in the Unity database. Alarms are raised from these logs (see "5.2 Events Data Translation") with a default priority level assigned to them by the switch. The alarm priority may be changed by an administrative user if required. You may, for example, need to change the priority of an alarm when it appears for the first time. You do this by reassigning the priority via the Unity Events Interface.

All alarms raised from Events data are assigned a common source ID.

Transmission Alarms

Unity generates alarms if any errors arise from the transmission of data from the Switch to Titan. There are three possible sources for these alarms, the CDR data transfer, the Events data transfer and the Statistics data transfer.

Each of these three alarm sources is assigned its own source ID.

Internal alarms

Not only may the modules themselves, generate alarms, but the links between modules may also generate alarms. Every module, sub-system, and link that makes up Unity may generate an alarm, for which it assigns a unique alarm source ID. This makes it much easier to track possible problems within the Unity system.

NOTE:

You may only change the priority of external alarms. You do this via the Unity Events Interface. You cannot change the priority of internal alarms.

7.2 Dealing with Alarms

Once alarms have been raised, they need to be dealt with. The purpose of an alarm is to bring the user's attention to a problem with the system. The problem may be minor and dealt with very easily and quickly, or it may be major where the solution is more difficult. In any case, once the problem has been solved, the alarms can be cleared by a user with sufficient security clearance, usually the system administrator. The Unity Alarm Interface (UAI) is the front-end that the user uses to deal with alarms. This is described fully in the accompanying UIMS manual

Unity handles existing or potential problems by raising alarms and providing the means of issuing Trouble-tickets. Each time an alarm is activated, you have the option to create a Trouble-ticket. A Trouble-ticket is a means by which you can track the rectification process of a particular problem. This facilitates successful problem management, and the feature should be used wherever possible. You manage Trouble-tickets using the Unity Trouble-tickets Interface, which is accessible from the Unity Interface Manager. Interface. Trouble-tickets are fully explained in the accompanying UIMS manual.

NOTE:

If you have the optional Unity Pager Interface, you can page somebody in the event of certain alarms. The alarms that trigger a pager and the user who is paged can be set up in the Unity Pager Interface.

8 CDR Data File Processing

Part of the Diamond solution is the existing CableMaster billing system and the FMS. Although it would be possible to use a billing system that utilises the Unity database tables, a solution presently exists whereby a copy of the raw CDR data file is formatted for and processed by Cablemaster. This chapter describes how this is achieved, and details the directory structures, and files used in the CDR transfer process, as well as their naming conventions. This chapter consists of the following sections:

- 8.1 General Description of CDR Data Transfer
- 8.2 Directory Structures
- 8.3 CDR File Naming Convention

8.1 General Description of CDR Data Transfer

Figure 3 shows how NTL's existing billing system interfaces with Unity. The CDR data file downloaded from the Switches are simply copied into other directories on Titan where they are then formatted accordingly and passed onto the CableMaster billing system and the FMS system. Figure 4 shows the CDR data-flow in more detail for the different Switch types, i.e. System-X and DMS.

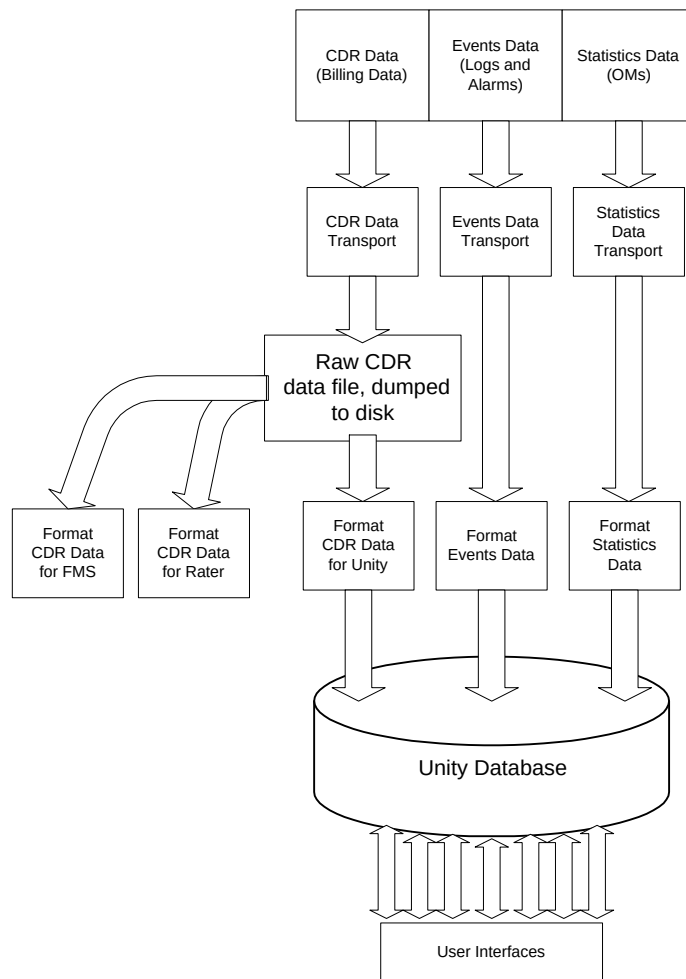


Figure 3: Unity Data-flow with Existing Billing System for one of the Switches

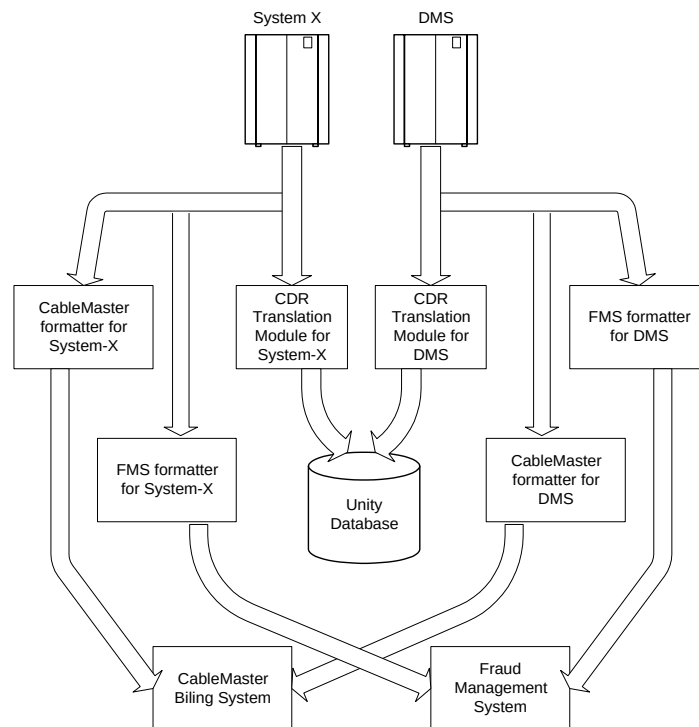


Figure 4: CDR Data-flow for different Switch types.

8.2 Directory Structures

The following directory structure is placed on the Unix server:

NOTE:

The directory structures and filenames are specific to Diamond and do not apply to other versions of Unity.

8.2.1 The Directory: /products/unity

This is the home directory for the CDR transfer. This resides on a logical disk and is parent to all of the following sub-directories.

The Sub-directory: /products/unity/bin

This sub-directory contains all the necessary executable files and scripts.

WARNING!

None of the files in the above directories are to be edited by the customer. Only Dataphone personnel are qualified to manipulate these files. Unauthorised usage may cause loss of synchronisation and could even result in the loss of billing records from the Switch or the Unity server.

The Sub-directory: /products/unity/LOGS

This directory contains the log files created during individual transfer and formatting cycles. There are several different types of log files, depending on

the Switch type and the format type, but are generally of the form; <operation>_yyjjj_PX.log where **yy** is a two digit year number and **jjj** is a three digit number representing the day of the year in the Julian format, and X is the poll sequence. The following examples should clarify this:

Examples:

The following log files were created on the 212th day of 1998, i.e. 31 July 1998:

sysxfr_99212_P2.log is a log file concerning the *transfer* of a file of *poll sequence number 2* from a *System-X* Switch.

dmsxfr_99212_P3.log is a log file concerning the *transfer* of a file of *poll sequence number 3* from a *DMS* Switch.

fmsfmt_99212.log is a log file concerning the *fms formatting* of a file of *poll sequence number 4*.

These log files contain all the information needed to trace the mediation operation. Some of the essential information contained in these log files are also stored in the Unity database for analysis and presentation. You can view this log file directly by transferring a copy to a PC and importing it into a standard ASCII editor such as windows Notepad.

The Sub-directory: /products/unity/etc

This directory contains the files, CRONTAB.on, CRONTAB.off, config.dat, sysxfmt_seq.dat, sysxpoll_seq.dat and dmsspoll_seq.dat. The **CRONTAB** files control the automated CDR operation, config.dat contains the configuration commands for Unity, and the ...seq.dat files contain the poll sequence number for the various operations.

If you wish to terminate the automated poll, use the following command:

```
crontab CRONTAB.off
```

To restart the process, enter:

```
crontab CRONTAB.on
```

WARNING!

If the automatic polling sequence is stopped, it will not be resumed until it is manually switched on again, this time using the CRONTAB.on file as the argument. The crontab command should only be used under the direction of Dataphone technical support.

The Sub-directory: /products/unity/audit

This directory contains a sub-directory for each of the Switch sites (Table 1), where the audit file for the respective Switch is held.

Each of the site specific directories contains further sub-directories /old, /older and /oldest. When a newer audit log file is produced, the existing one is transferred to the sub-directory /old. The one that is in /old

is transferred to /older, and the one that was in /older is transferred to /oldest. The old audit file in /oldest is then lost. This is the only form of backup for the audit files as they can easily be reproduced from existing backed up log files.

The Sub-directory: /products/unity/lock

This directory contains locks for the Switch that is currently being polled. These locks are an essential part of the polling synchronisation, and are not to be tampered with under any circumstances.

The Sub-directory: /products/unity/dmstmp

This directory is used for holding the intermediate raw DMS data to be formatted before it is transferred to its final location.

The Sub-directory: /products/unity/sysxtmp

This directory is used for holding the intermediate raw DMS data to be formatted before it is transferred to its final location.

8.2.2 The Directory: /sites

This is a symbolic link (shortcut) to the main directory /export/home/ama, where the data for each respective site is held in site-specific directories.

8.2.3 The Directory: /export/home/ama

This directory contains the data that is retrieved from the Switches on a regular basis. It occupies an entire logical partition and is typically of the order of a few Gigabytes. The exact size of the partition is dependent on the estimated quantity of data retrieved from the Switches and how long it is to be held before archiving or deleting.

Each Switch is assigned it's own sub-directory on this partition. The sites are named as follows:

Directory Name	Description	AS/400 Site ID	Site Code
dalee	DMS at Daleside	DSE	E
dalea	System-X at Daleside	DSA	A
daleb	System-X at Daleside	DSB	B
dalec	System-X at Daleside	DSC	C
daled	System-X at Daleside	DSD	D
oswin	DMS at Oswin Road	OSW	O
shepshed	DMS at Shepshed	SHE	S
northfld	DMS at Northfield	NOR	N
wood	System-X at Woodborough Road	WOO	W
mansfld	System-X at Mansfield	MAN	M
grimsby	System-X at Grimsby	GRI	G
lincoln	System-X at Lincoln	LIN	L

Table 1: Site Names

Each site sub-directory is divided to four further sub-directories: /live, /raw, /bkup and /audit.

Examples:

The sub-directory /sites/dalee/raw contains the raw data retrieved from the DMS at Daleside Road East.

The sub-directory /sites/wood/live contains data corresponding to the System-X at Woodborough Road, formatted for the AS/400 billing system.

The sub-directory /sites/oswin/bkup contains files from the DMS at Oswin, that have already been sent to the AS/400, and are now due to be backed up before deleting.

The sub-directory /sites/shepshed/audit contains the audit trail per file polled for the DMS at Shepshed.

NOTE:

While the files are in transitional state, they are hidden and are not visible.

8.2.4 The Directory: /export/home/ama/fms/<site>

This is the home directory for the files to be used by the FMS. Like the sites directory, this directory has a sub-directory for each Switch.

These sub-directories contain the files that have been formatted for the FMS. It is the responsibility of the FMS to move these files regularly to their own processing system. This responsibility lies with the FMS for security reasons.

WARNING:

Any files in the fms directory that are older than seven days will be automatically deleted! It is therefore important that the FMS has means to retrieve the files on a regular basis and carry out its own backup procedure.

In case of emergencies, it is possible to reconstruct the FMS files from backed up raw data. Should this situation arise, this operation may only be carried out by Dataphone personnel.

8.3 CDR File Naming Convention

The lengths of the filenames are limited to 10 characters. this is because the IBM AS/400 cannot handle filenames that exceed this character length.

The CDR files naming conventions are **u_yyjjjspa**, where:

u_	designates an unprocessed file
yy	designates the year number, for example 98
jjj	designates the Julian day of the year. For example 212 for 31st July
s	Switch Code: For example, E for Daleside East.
p	is the alphanumeric poll sequence number. This is only meaningful if the automated system is in place. The poll sequence synchronises itself at 00:00 hours, such that the first poll of the day is poll number 1, the second poll is 2 etc. The poll sequence is reset to 1 every midnight.
a	is an alphabetic character indicating the file sequence. This is to ensure that incoming files from the switch in the same polling session do not overwrite each other. The first file is designated with an uppercase A character, the second file with a B character, and so on until all 26 uppercase characters have been used up. The 27th file is designated with a lower case a character, and so on. There is therefore a capacity of 52 files. In the normal case there will be no more than a few files, but in the case of testing there may be significantly more. If more than 52 files are produced in one session, the sequence wraps around and starts with A again. Therefore 52 files is the maximum amount of files that can be processed in one session.

Examples:

The files: u_99212S1A and u_9212S1B indicate the first polled files of the day, retrieved from the Switch at Shepshed.

The first polled file for the year 2000, will be u_00001S1A.

The formatted files for the FMS have the above name but with the .fmt extension.

8.4 The Processes

There are four main processes involved in a CDR polling operating cycle. The processes utilise the UNIX multi-tasking capabilities to increase the efficiency of the procedure. The four processes, which can run sequentially, concurrently, or a combination of both, are summarised below.

Process 1 (Retrieval Process)

At regular intervals, Unity polls the Switch and receives the raw CDR data. The individual files are then transferred from the Switch to the Unity server, over the X.25 network. The data are stored in a directory that has been

assigned to the particular Switch, i.e. /export/home/ama/<site name>. This process also generates the transfer log files and places them in the directory /products/unity /LOGS. This log file is dynamic and is being appended continually as the retrieval process continues.

The exact length of the process depends on several factors such as the quantity of raw data to be retrieved, the data transfer rate, and the Switch response time, but is typically of the order of tens of minutes. The process repeats itself on a regular basis (e.g. hourly).

Process 2 (Formatting Process)

After all the files have been retrieved from the Switch, a second process is automatically initiated to format the data retrieved from the Switch. The raw data are formatted into two formats. One for the SMS, for billing purposes, and one for the FMS. The AS/400 formatted data are placed in the directory /export/home/ama/<site name>/live, and the FMS formatted data are placed in the directory export/home/ama/fms/<site name>. This process also generates the format log file and places it in the directory /products/unity/LOGS. This log file is dynamic and is being appended continually as the formatting process continues.

The length of this process is of the order of minutes..

Process 3 (Auditing Process)

After the data for Switch 1 has been formatted, a third process is initiated. This process takes a snapshot of the two dynamic log files that have been generated by the first two processes, and then produces an audit file for the respective Switch. This audit file is called adt.log. and is stored in the directory /sites<site name>/audit.

Example: /products/unity/audit/dalee/u_yyjjcsi.adt

The length of this process is of the order of seconds.

Process 4 (Database Loading Process)

The CDR data is loaded into the CDR table in the database

Figure 5 depicts the flow of the various processes for sequential polling, and Figure 6 depicts the various processes for concurrent polling.

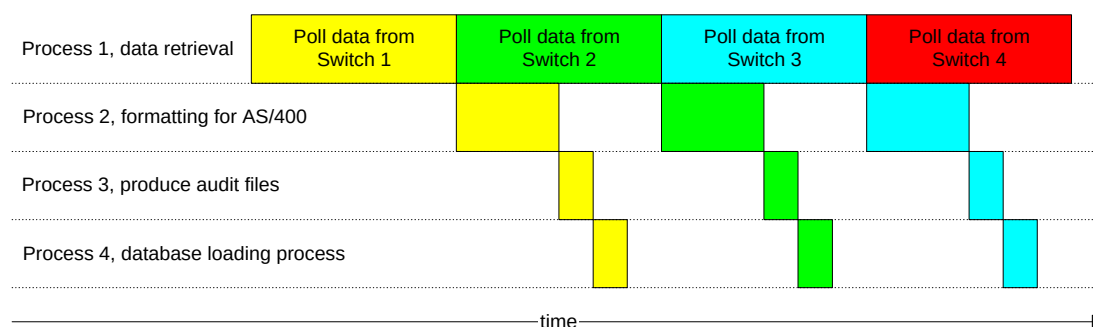


Figure 5: Sequential Data Retrieval and Processing

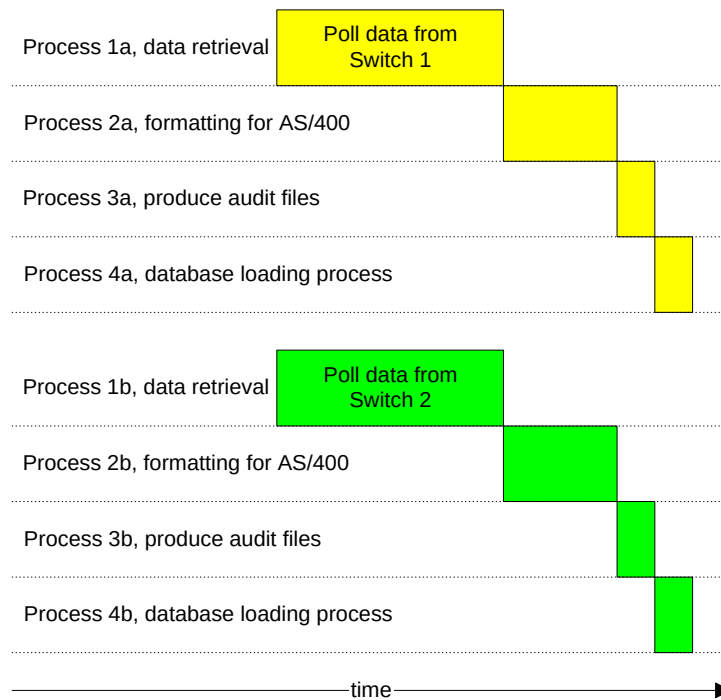


Figure 6: Concurrent Data Retrieval and Processing

The system can be set up for concurrent or sequential polling, or a combination of both, depending on which combination is more efficient in terms of data availability. For instance, if the polling process for one Switch takes a long time (say a couple of hours), whilst the other Switches only take a few minutes each, it would be advisable to poll this Switch in parallel to the others (which may be being polled sequentially with respect to one another). Dataphone will ensure that the optimum polling process is set up.

After all the various processes are finished, the data are in formats suitable for billing purposes, and for the FMS. Once used, these data are held in the system for seven days in order to allow an audit to be carried out on historic data, after which they are deleted.

9 Maintenance

The system is maintained by Dataphone according to the contractual agreement. There are however occasions when a qualified system administrator or operator could carry out certain administrative tasks. The following sections summarise the automatic file management processes, the possible fault conditions, error messages and remedial actions. This chapter consists of the following sections:

- 9.1 Manual File Management
- 9.2 Automatic Backup Procedure
- 9.3 Hard Disk Capacity
- 9.4 Manual Cleanup Tasks

9.1 Manual File Management

The unprocessed files designated with a with a preceding `u_` are placed in the relevant System-X or DMS `live` directory. The AS/400 regularly checks this directory and picks up the files for processing. After the AS/400 has retrieved all the formatted data from this directory it must rename the original files, replacing the `u_` (for unprocessed) with a `b_` (for backup).

For example, the file `u_98264E1C` must be renamed to `b_98264E1C`.

NOTE:

It is imperative that the AS/400 system carry out this task, otherwise the clean-up operation will not work.

Titan regularly checks for `b_` files in the `live` directory and moves them to the `bkup` directory where they are copied to DAT tape during the nightly tape backup process.

9.2 Automatic Backup Procedure

The automated file manager regularly checks for backup files and moves them to the location `/sites/<site name>/bkup`. These are later backed up onto tape and deleted from the hard disk.

A clean-up operation is carried out every night. All critical directories are searched and all files older than 7 days are backed up to a DAT tape and then deleted.

A further clean-up operation takes place automatically once a week. This is initiated by a file manager application on Titan. In this case, the files in all of the backup directories are backed up to a DAT tape and then compressed and moved to a holding directory for a further seven days. You must make sure that a blank clean tape is inserted in the DAT drive every Saturday night

The time periods are configurable if required. Please contact Dataphone technical support if you wish to redefine these time periods.

9.3 Hard Disk Capacity

The hard disk capacity should be checked regularly (at least once a week). If the hard disk gets too full, you may experience errors. This could happen if either the automated tape backup or the automated file management process has failed. This in turn may be due to the DAT drive not being operational (for example, being switched off, no tape in the drive, disconnected etc.). To remedy this situation, do the following:

1. Ensure the tape drive is fully operational and that a blank writeable tape is inserted.
2. Log onto Titan using the Username root and appropriate password. You will be prompted by the hash sign '#' after successful log on.
3. Carry out the procedure described below to free up some hard disk space.

Procedure to Release Hard Disk Space

Execute the system command `df -k`. This command produces information pertaining to the hard disk. Look for the partitions `/products`, `export/home`, and `/sites` in turn, and note the percentage used under the '%used' column. If this figure is more than 90%, insert a blank DAT tape in the tape drive and carry out the following commands at the # prompt and follow them by a carriage return:

1. `cd /products/unity/bin`
2. `./dcc_fmanage` (this may take several minutes)

This operation should back up all files more than seven days old to the DAT, and then delete them from the hard disk.

Now recheck the percentage used, by executing the `df -k` command once again. The percentage used should now be less than 75%. If it is not, then contact Dataphone technical support.

9.4 Manual Cleanup Tasks

Mail system

The system administrator should check the Unix mails on a regular basis. Unwanted mails should be deleted. If in doubt, please call Dataphone for advice.

Delete Old Log Files

The support staff should check the CDR transfer logs and format logs regularly to monitor smooth and fault-free operation. These files should be backed up for future reference and then deleted from the server. If the log files report any problem, you should contact Dataphone immediately.

Register Unknown Events (Unity User Interface Operation)

It is very important to register new events, as and when they arrive. This is especially the case when the Unity installation is quite young, as a lot of the events originating from the Switch will be unknown. As Unity matures, its library of known events increases and the number of new unknown events will reduce. If unknown events are never registered, the number of records stored in the Unknown Events Queue will simply increase and eventually

overload the system. It is easy to register unknown events. Event handling is described fully in the Unity User Guide.

Clear Old Alarms (Unity User Interface Operation)

It is also very important to clear alarms, after they have been dealt with. The automatic Unity Cleanup Operation is not allowed to permanently delete alarms until they have first been cleared. If you do not clear alarms, the Alarm Queue will simply increase in size, slow down the performance, and eventually the system will get overloaded. Dealing with and clearing alarms is fully described in the Unity User Guide.

10 Disaster Recovery

This chapter describes the procedure that must be followed in order to avoid or minimise the loss of call data from the Switches in the event of operation failure. This chapter consists of the following sections:

10.1 Mediation System Normal Operation

There are four major components in the normal working system:

The General Unity Server Titan

A SUN Server as described in "2.1 Overview of Infrastructure"

The Switches

The Switches themselves are also relevant in the DR system, in that they may require human intervention as part of the process.

The Communications Links

There is a frame relay network using ONI equipment connecting Titan at Daleside Road with each switch. The protocol used is X.25 with XFER for DMS and X.25 with OSI for System-X. This network has sufficient redundancy to work normally in the event of a single failure.

The Billing Systems

Cablemaster and the Fraud System retrieve billing data via Ethernet using FTP. Rater retrieves data via an NFS mount to an intermediary Linux PC.

Normally, Titan polls call data from each Switch at regular intervals defined by the system. For Diamond, this is normally every hour. These raw data are temporarily stored in the system, which employs mirrored disks for first level backup purposes. The data are then formatted and copied to the Unity Database and other locations where they await collection by the billing and fraud systems. The criticality of the data on the two different types of Switches is different, as described below.

System-X Switches

The DR system for data polled from System-X Switches is critical because once the data have been polled, they are immediately deleted and can never again be recovered. This means that if the polled data are lost due to disaster before they are backed up or picked up by the billing system, they are permanently lost. This likelihood is significantly reduced by mirroring the disks (in case of disk failure), but the risk cannot be completely eliminated (for example, fire or flood).

DMS Switches

DMS Switches are not quite as critical as System-X, in that once the data have been polled from the DMS they are not deleted but rather "marked for deletion". This means that it is possible to recover the data even after a total loss of Titan, however, the period in which the data may be recovered is unknown and should be assumed to be small.

10.2 Main Elements of the Disaster Recovery System

The essence of the Disaster Recovery System relies upon a second server called Pheobe (which resides in a different geographical location) and a series of procedures. Phoebe is a Sun E450 system, similar to Titan but with less capacity and without the Oracle Database. These procedures were developed in view of several worst case scenario assumptions:

1. The System-X Switches can only store data for a maximum of 24 hours, thus making it unacceptable for a Switch not to be polled on a complete working day.
2. System-X switches do not write reliably to tape devices. Tape backups should therefore be regarded as the absolutely final backup.
3. The use of tapes should be avoided as it is difficult to handle tapes of any sort for billing data.
4. In the event of a real disaster, System-X data may be lost.
5. In the event of a simple failure of part of the system, no data will be lost.
6. In the event of a system failure that requires the invocation of Pheobe, the only DR functionality required is that of sending data to the FMS, Cablemaster and Rater.
7. In the event of a complete disaster, no functionality is necessary except for the collection of data by Cablemaster, Rater and FMS.

As an additional warning in the near future, alarm systems will be made operable, and NTL's Regional Monitoring Centre, RMC will raise the alarm if any System-X Switch exceeds 10 MB of stored data.

Disaster Recovery Definition

The normal definition of *Disaster Recovery* is the ability to resume normal operations as soon as possible, with minimum disruption and loss of data in the event of catastrophic failure. In the context of this document, we extend this definition further to deal with events that are not quite a disaster in the strict sense of the word, but which *could* result in loss of data if prompt action is not taken. For example, a spare part might take so long to be delivered that there is a risk that there may not be enough time to install it and get Titan back up and running before the Switch starts to overflow. There are four grades of *Disaster* in the scope of this document:

- Very Severe - Total and permanent loss of Titan plus major communication problem.
- Severe - Total and permanent or long term loss of Titan
- Hot Standby - Temporary Loss of Titan (e.g. Repair taking several hours)
- Major Communication problem - Loss of communication, that cannot be resolved in the time available.

Each disaster grade is now described in more detail and an outline procedure is described. For further details of the individual steps involved, please contact Dataphone technical support.

10.3 Outline of procedures

Very Severe

It is possible that a disaster such as a fire in the computer room could cause destruction of Titan and the communications equipment. In the case of Titan follow the procedure for a Severe DR, and in the Communications case, follow the procedure for a Major Communications Problem.

Severe

This is the procedure that you should follow in the event that Titan is lost permanently or will be out of action for a long time (e.g. titan has to be taken away for repair).

1. Disable the crontab on Titan (if Titan still exists!)
2. Set the correct polling sequence on Phoebe
3. Start Phoebe Polling
4. Enable Crontab on Phoebe
5. Ensure switches are being polled and file numbers reduced
6. Collect any live files from Titan using AS/400 if connection available*
7. Recover any data still on Titan to Phoebe
8. Configure Cablemaster to collect data from Phoebe
9. Configure Rater to collect data from Phoebe
10. Configure Fraud to collect data from Phoebe

After titan has been rebuilt and reinstalled:

1. Set correct polling sequence on Titan
2. Test Poll from Titan (Carried out by support team)
3. Disable crontab on Phoebe
4. Start Titan Polling
5. Enable Crontab on Titan
6. Collect any live files from Phoebe using AS/400 if connection available*
7. Ensure remaining data collected from Phoebe
8. Configure Cablemaster to collect data from Titan
9. Configure Rater to collect data from Titan
10. Configure Fraud to collect data from Titan
11. Choose start data for Oracle Database
12. Re-install old data into Oracle

Hot Standby

This is the procedure to carry out in the event that you expect that Titan will be back on-line relatively soon, but it may take a time that is long enough to result in a risk of data overflow on the Switches. An example of this might be, for example hardware failure (e.g. CPU) where the spare part is taking substantially longer than expected to arrive. Clearly it would be prudent to start polling the switches from Phoebe in the meantime. However, there is no need to change the settings on systems such as Cablemaster for such a short period.

1. Ensure Titan is not polling and will not start to poll if the fault is repaired by a third party.
2. Start Phoebe Polling
3. Ensure switches are being polled and file numbers reduced
4. Fix Titan
5. Stop Phoebe Polling
6. Test Poll from Titan (Carried out by support team)
7. Start Titan Polling
8. Move Data from Phoebe to Titan with appropriate filenames

Major Communications Problem

This is the procedure to be carried out in the event that there are communications problems resulting in the likelihood that one or more switches will not be polled for such a time as they may fill up. You can calculate the approximate time available from past records. For a busy System X switch this may be only a matter of hours whereas for a DMS or a quiet System-X it could be several days.

There are then two options:

1. Retrieve data from the Switch backup tapes
2. Physically move Phoebe to each site and poll that switch locally using a spare packet switch.

Glossary

Term	Description
CDR	Call Detail Recording
DAT	Digital Audio Tape
DIRPHOLD	Device Independent Recording Package Hold
DMS	Digital Multiplex System (Nortel type of Switch)
DTX	Digital Telephone Exchange
FMS	Fraud Management System
FTP	File Transfer Protocol for Internet
LAN	Local Area Network
OM	Operational Measurement (Statistical Data)
OSI	Open Systems Interface
PAD	Packet Assembler / Disassembler
SCSI	Small Computer Systems Interface
SMS	Subscriber Management System
SYSX	GPT System-X switch.
TCP/IP	Transfer Control Protocol / Internet Protocol
UNIX	An operating system
VI	The UNIX editor
WAN	Wide Area Network
X.25	A CCITT data communications protocol for WANs

Table 2: Glossary of Abbreviations

Term	Description
CDR Data	Generic term for the operational data sent from the Switch that is ultimately used for billing purposes. CDR stands for Call Detail Record, and may be of many different formats.
Billing Data	Another term for CDR data.
Event	An occurrence of something for which the Switch sends a warning message. The event may or may not be a problem. In many cases it is just informative. Events data are streamed to a specific port on the Switch.
Alarm	A record of a <i>problematic</i> Event that has a non-zero priority level
Statistical Data	Generic term for operational measurements carried out at the Switch. Like Events data, Statistical data are streamed to a specific port on the Switch.
Operational Measurement	A DMS term for a statistical data record.

Table 3: Special Terms

Index

- Alarm Management..... 19
- Alarm Source ID 19
- Alarms
 - definition 3, 35
 - external 19
 - internal 19
 - raising 20
 - sources 19
- Auditing Process..... 28
- Automatic Backup Procedure 31
- Billing Data..... See CDR Data
- CableMaster 5
- Call Record Data See CDR
- Call records..... 13
- CDR 3, 35
- CDR Data 3, 10, 13, 35
 - filename convention 26
- CDR Data File Processing 21
- CDR Data Transfer 21
- CDR Data Translation 15, 17
- CDR Transport..... 13
- Concurrent Data Retrieval 29
- CRONTAB 24
- DAT..... 35
- Data
 - flow diagram 10
 - formatting..... 11
 - packets 14
 - raw 13
 - transfer..... 13
- Data components
 - CDR data 10
 - Events data 10
 - pictorial representation..... 10
 - Statistical data 11
- Data Storage..... 17
- Database
 - interfacing 17
 - tables 17
- Data-flow..... 9
- Directory Structures 23
- DIRPHOLD 35
- DMS..... 21, 35
- Double Server Failsafe Operation.... See DSFO
- DSFO..... 6
- DTX..... 3, 10, 13, 35
- Event..... 3, 35
- Events Data 10
 - Translation 15
 - Transport 14
- File Management 31
- File Transfer Process..... 28
- FMS 35. See Fraud Management System
- Formatting Process..... 28
- Fraud Management System..... 21
- Front Ends See User Interfaces
- FTP 35
- Glossary 35
- Hard Disk Capacity..... 31
- Help system
 - main 11
 - Whats This 11
- LAN 35
- Layers
 - pictorial representation 10
- Log 3
- Log files..... 23
- Maintenance..... 31
- Manual Cleanup Tasks..... 32
- Mediation..... 13
- Mediation Infrastructure..... 5
- OMs..... See Operational Measurements
- Operational Measurements 3, 10, 35
- Oracle..... 11, 17
 - advantages..... 17
- OSI 35
- PAD 6, 35
- PAD emulator 11, 14
- Polling interval 13
- Retrieval Process 27
- Rotation 13
 - Switch data..... 13
- SCSI..... 35
- Sequential Data Retrieval..... 29
- Statistical Data 3, 11, 35
 - Translation 16
 - Transport..... 14
- Statistical Data Translation..... 17
- Switch..... 3
- System-X..... 21
- TCP/IP..... 35
- Translation Layer..... 11, 15
- Transport Layer 11, 13
- Trouble-tickets..... 19
- UALARMS..... 18
- UBILLINGSTATUS..... 18
- UCDRDATA 18
- UNKNOWNLOGS 18
- ULOGS..... 15, 18
- Unity Alarm Interface..... 20
- Unity Database..... 11, 17
- UNIX..... 35
- URETENTION 18
- User Interfaces..... 11
- USOURCE_ID..... 18
- USWITCHLIST 18
- UTIMECHANGE..... 18
- UUNKNOWNLOGS..... 18
- UUSERS 18
- WAN..... 35
- X.25..... 35
 - card 14
 - network..... 13